CTF-CRYPTO-RSA-Evaluate

原创

大熊何在 于 2020-11-27 17:24:40 发布 53 收藏

分类专栏： CRYPTO

本文链接：https://blog.csdn.net/zippo1234/article/details/110238017

CTF-CRYPTO-RSA-Evaluate

每天一题，只能多不能少

RSA-Evaluate

题目分析

1. 爆破x和y
2. 数学计算

开始

1.题目

```
n = 240152805134947292476871478328219119218832547749382061299563916172003651729128615218824774769469250174112317
232907232890534755807392423532568620406113713781934448944302608229687693011765158936208313349932447280437358796457996156161596733087581540571970724616038709204452827867105118443705180348733048399133837556546250435246793925780572388600627820997
e = 65537
((y%x)**2018+(y%x)**2019+(y%x)**2020)**2+(x**365)+(y/x)**2 = 484827245275013618580814698980966335448136775417779259022187221949836650550054180024853667857329197335088101448654487877657910925540048916339993473755627999318583039929451613148172411379379888307287737276438361583112411891837361830271605083763828866352062182677651590830626339209815123614263478318098919908875636776577780944021114221647882724366574274742822519577071638663184603199045000810406787134269285828705040919422556542052409279234755131073553174150699026316229049969788099434695572039895712611211562086811275870141866824393612643090346057893628497670122819695305316895300870169642862628383092510332332566161796549162002251273686246169879721291300493369011717083769817804184105278624103853426089837323874032674339471209560980921
p=gmpy2.next_prime(x*y*z)
q=gmpy2.next_prime(z)
c = 654358225977792758323477081930063181340286962867905170163169413844202012581254893399154696020859327229507304801309287969527245302711873152617008207806037369322851054673530185156749733531907083600436591048017704174314723536149310134537755882391987634686392994232551401653935451150622724001529852124177015027973140985857898346024349197291073701043667 07
```

2.分析

和某次CTF比赛的题目类似。这题折腾了我好几天，实在惭愧，难度不大。
先爆x的范围

```
for x in range(1000):
    if x**365>A:
        y-=1
        break
```

可以得到x最大是103
根据经验判断y%x只能是0或者1。
则有

```
for x in range(1,104):
    for y_x in range(2):
        if iroot(A-((y_x)**2018+(y_x)**2019+(y_x)**2020)**2-(x**365),2)[1]:
            print x,y_x
```

得到x就是103，y%x为1。
那就很简单了。

3.脚本

```python
#!python2
# -*- coding: utf-8 -*-
# @Time : 2020/11/27 11:24
# @Author : A.James
# @FileName: t1.py
from gmpy2 import *
from libnum import *

n = 24015280513494729247687147832821911921883254774938206129956391617200365172912861521882477476946925017411231723290723289053475580739242353256862040611371378193444894430260822968769301176515893620831334993244728043735879645799615616159673308758154057197072461603870920445282786710511844370518034873304839913383755654625043524679392578057238860062782099 7
c = 65435822597779275832347708193006318134028696286790517016316941384420201258125489339915469602085932722950730480130928796952724530271187315261700820780603736932285105467353018515674973353190708360043659104801770417431472353614931013453775588239198763468639299423255140165393545115062272400152985212417701502797314098585789834602434919729107370104366707
A = 48482724527501361858081469898096633544813677541777925902218722194983665055005418002485366785732919733508810144865448787765791092554004891633999347375562799931858303992945161314817241137937988830728773727643836158311241189183736183027160508376382886635206218267765159083062633920981512361426347831809891990887563677657778094402111422164788272436657427474282251957707163866318460319904500081040678713426928582870504091942255654205240927923475513107355317415069902631622904996978809943469557203989571261121156208681127587014186682439361264309034605789362849767012281969530531689530087016964286262838309251033233256616179654916200225127368624616987972129130049336901171708376981780418410527862410385342608983732387403267433947120956098092 1
e=65537

for x in range(1,104):
    for y_x in range(2):
        if iroot(A-((y_x)**2018+(y_x)**2019+(y_x)**2020)**2-(x**365),2)[1]:
            print x,y_x
y=iroot(A-((y_x)**2018+(y_x)**2019+(y_x)**2020)**2-(x**365),2)[0]*x+y_x
print y
z = iroot(n/(x*y),2)[0]
print z
q = next_prime(z)
p = n//q
phi = (p-1)*(q-1)
d = invert(e,phi)
m =pow(c,d,n)
print n2s(m)
```

4.get flag

```
flag{EvaLuaT3_raNges}
```

结语

简单计算的问题。